

**In The Claims:**

1. (Currently amended) A method of dynamically deploying services in a computing network, comprising ~~steps of~~:

receiving client requests for a selected service, wherein the client requests are received from outside a local network that communicatively connects an origin server and one or more edge servers;

serving the received requests from ~~a first~~ the origin server when the selected service has not yet been dynamically deployed;

effecting a dynamic deployment by programmatically moving the selected service from the ~~first origin~~ server to the one or more other edge servers, which are closer to an edge of the local network than the origin server, when the dynamic deployment is triggered; and

serving the received requests from the one or more ~~other edge~~ servers after the ~~effecting step causes the~~ selected service has been ~~to be~~ dynamically deployed to the one or more edge servers.

2. (Currently amended) The method according to Claim 1, further comprising ~~the steps of~~:

monitoring a number of the received client requests for the selected service; and

triggering the dynamic deployment when the monitored number exceeds a predetermined threshold.

3. (Original) The method according to Claim 2, wherein the predetermined threshold applies to a plurality of dynamically deployable services.

4. (Original) The method according to Claim 2, wherein the predetermined threshold applies to the selected service.

5. (Currently amended) The method according to Claim 2, wherein a value of the predetermined threshold applies to all of the one or more ~~other~~ edge servers.

6. (Currently amended) The method according to Claim 2, wherein values of the predetermined threshold apply to individual ones of the one or more ~~other~~ edge servers.

7. (Original) The method according to Claim 2, wherein a value of the predetermined threshold is specified by a systems administrator.

8. (Original) The method according to Claim 2, wherein a value of the predetermined threshold is specified as a default value.

9. (Original) The method according to Claim 2, wherein a value of the predetermined threshold is specified programmatically.

10. (Currently amended) The method according to Claim 2, wherein the monitoring step counts the received client requests at individual ones of the one or more ~~other~~ edge servers.

11. (Currently amended) The method according to Claim 2, wherein the monitoring step counts the received client requests at a plurality of the one or more ~~other~~ edge servers.

12. (Currently amended) The method according to Claim 1, further ~~comprising the steps of:~~

monitoring a load on the local ~~computing~~ network; and

triggering the dynamic deployment when the monitored load exceeds a predetermined threshold.

13. (Currently amended) The method according to Claim 1, wherein effecting a dynamic deployment ~~the programmatically moving step~~ further comprises ~~the step of~~ issuing a deployment request for the selected service.

14. (Original) The method according to Claim 13, wherein the deployment request comprises a service description of the selected service encoded in a standardized service description notation.

15. (Original) The method according to Claim 14, wherein the service description comprises an interface definition of a dynamic deployment service and an implementation definition of the dynamic deployment service.

16. (Original) The method according to Claim 15, wherein the dynamic deployment service resides on the ~~first~~ origin server.

17. (Currently amended) The method according to Claim 13, further comprising ~~steps~~ of:

receiving the deployment request at a particular one of the one or more ~~other~~ edge servers, the particular one of the edge servers being that server to which the selected service is being dynamically deployed;

issuing a subsequent deployment request, responsive to receiving the deployment request at a particular one of the one or more edge servers ~~the receiving step~~, from the particular one of the edge servers to the ~~first~~ origin server; and

receiving, responsive to the subsequent deployment request, a deployment response from the ~~first~~ origin server.

18. (Original) The method according to Claim 17, wherein the subsequent deployment request comprises a SOAP ("Simple Object Access Protocol") request.

19. (Original) The method according to Claim 17, wherein the subsequent deployment request comprises an XML (“Extensible Markup Language”) Protocol request.

20. (Original) The method according to Claim 17, wherein the subsequent deployment request identifies the selected service.

21. (Currently amended) The method according to Claim 17, wherein the subsequent deployment request provides information about run-time conditions on the particular one of the edge servers.

22. (Currently amended) The method according to Claim 17, wherein the deployment response comprises executable code to be deployed on the particular one of the edge servers.

23. (Currently amended) The method according to Claim 21, wherein the deployment response comprises executable code which is adapted to the run-time conditions on the particular one of the edge servers.

24. (Currently amended) The method according to Claim 22, further comprising ~~the~~ step of deploying the executable code on the particular one of the edge servers.

25. (Currently amended) The method according to Claim 1, further comprising ~~the~~ steps of:

transparently routing the received client requests using a repository which tracks whether the selected service is deployed on the one or more ~~other~~ edge servers; and

wherein the serving steps serve the received requests from the ~~first~~ origin server or from the one or more ~~other~~ edge servers, depending on the ~~transparently~~ transparent routing of the received client requests ~~step~~.

26. (Currently amended) A system for dynamically deploying services in a computing network, comprising:

a local network that communicatively connects an origin server and one or more edge servers;

means for receiving client requests for a selected service, wherein the client requests are received from outside the local network;

means for serving the received requests from ~~a first~~ the origin server when the selected service has not yet been dynamically deployed;

means for effecting a dynamic deployment by programmatically moving the selected service from the ~~first~~ origin server to the one or more other edge servers, which are closer to an edge of the local network than the origin server, when the dynamic deployment is triggered; and

means for serving the received requests from the one or more ~~other~~ edge servers after the ~~effecting step causes the~~ selected service has been to be dynamically deployed to the one or more edge servers.

27. (Currently amended) A computer program product for dynamically deploying services in a computing network, the computer program product embodied on one or more computer-readable media and comprising:

computer-readable program code that is configured to receive ~~means for receiving~~ client requests for a selected service, wherein the client requests are received from outside a local network that communicatively connects an origin server and one or more edge servers;

computer-readable program code that is configured to serve ~~means for serving~~ the received requests from ~~a first~~ the origin server when the selected service has not yet been dynamically deployed;

computer-readable program code that is configured to dynamically deploy ~~means for effecting a dynamic deployment~~ by programmatically moving the selected service from the ~~first~~ origin server to the one or more other edge servers, which are closer to an edge of the local network than the origin server, when the dynamic deployment is triggered; and

computer-readable program code that is configured to serve ~~means for serving~~ the received requests from the one or more ~~other~~ edge servers after the ~~effecting step causes the~~ selected service has been to be dynamically deployed to the one or more edge servers.